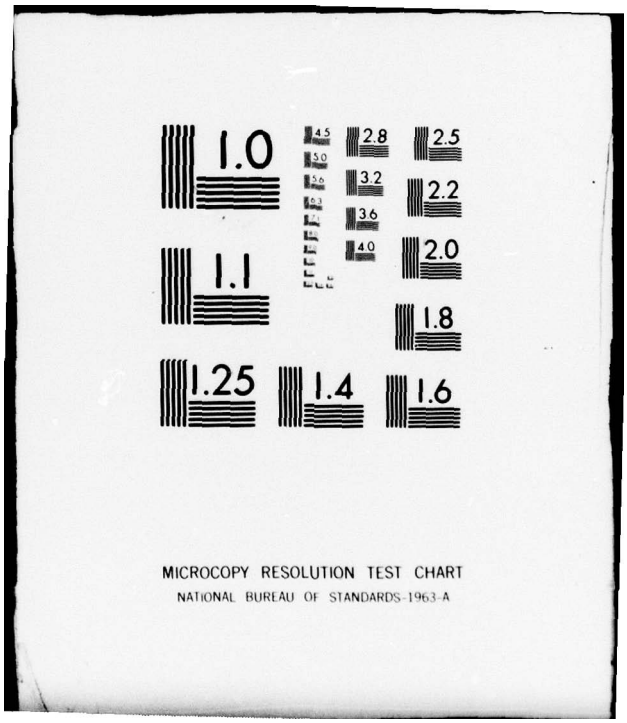
AD-A079 719 WISCONSIN UNIV-MADISON MATHEMATICS RESEARCH CENTER F/G 12/1
COMPUTING OPTIMAL LOCALLY CONSTRAINED STEPS.(U)
OCT 79 D M GAY DAAG29-75-C-0024
UNCLASSIFIED MRC-TSR-2013 NL

1 OF 1
AD-
A079719

END
DATE
FILMED
2-80
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(9) MRC Technical Summary Report # 2013

(6) COMPUTING OPTIMAL LOCALLY CONSTRAINED STEPS,

(10) David M. Gay

LEVEL II

Mathematics Research Center
University of Wisconsin—Madison
610 Walnut Street
Madison, Wisconsin 53706

(11) October 1979 (12) 34 (14) MRC-TSR-2013

(Received July 26, 1979)

(15) DAAG29-75-C-0024, NSF-MCS78-09525

Approved for public release
Distribution unlimited

Sponsored by

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park
North Carolina 27709

National Science Foundation
Washington, D.C. 20550

221 200

LB

UNIVERSITY OF WISCONSIN-MADISON
MATHEMATICS RESEARCH CENTER

COMPUTING OPTIMAL LOCALLY CONSTRAINED STEPS[1]

David M. Gay

Technical Summary Report # 2013
October 1979

ABSTRACT

In seeking to solve an unconstrained minimization problem, one often computes steps based on a quadratic approximation q to the objective function. A reasonable way to choose such steps is by minimizing q constrained to a neighborhood of the current iterate. This paper considers ellipsoidal neighborhoods and presents a new way to handle certain computational details when the Hessian of q is indefinite, paying particular attention to a special case which may then arise. The proposed step computing algorithm provides an attractive way to deal with negative curvature. Implementations of this algorithm have proved very satisfactory in the nonlinear least-squares solver NL2SOL.

AMS(MOS) Subject Classifications: 90C30, 65K05

Key Words: Unconstrained optimization, negative curvature

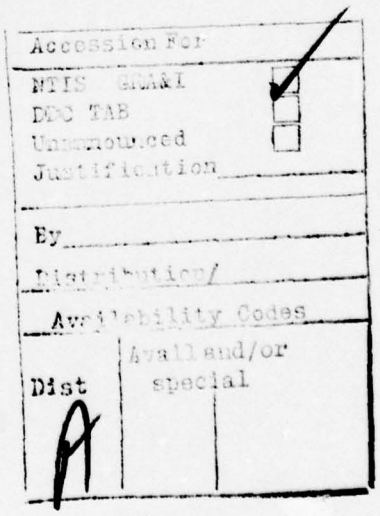
Work Unit Number 5 - Mathematical Programming and Operations Research

1 Presented at the Tenth International Sympsoium on Mathematical Programming, Montreal, Canada, 27-31 August 1979.

Sponsored by the United States Army under Contract No. DAAG29-75-C-0024. This material is based upon work supported by the National Science Foundation under Grant Nos. MCS78-09525 and MCS76-00324.

SIGNIFICANCE AND EXPLANATION

Unconstrained minimization problems arise in many contexts. Thus, given an objective function $\varphi : \mathbb{R}^n \to \mathbb{R}$, one often must seek a point x^* which minimizes $\varphi(x)$. It is usually necessary to resort to some iterative procedure: one computes a sequence of iterates $x_1, x_2, \ldots$ which, if all goes well, converges to x^*. Given the current iterate $x = x_k$, one commonly uses a quadratic approximation q to φ in computing the new iterate $x_+ = x_{k+1}$. Since q may only be accurate on a neighborhood N of x, one appealing way to compute x_+ is by minimizing q on a specified N. This paper deals with ellipsoidal neighborhoods N and presents a new way to handle certain computational details when the Hessian of q is indefinite (i.e., not positive definite). The resulting algorithm provides an attractive way to compute x_+ even when q is not convex. Modules implementing this algorithm give very satisfactory performance in the non-linear least-squares solver NL2SOL.



The responsibility for the wording and views expressed in this descriptive summary lies with MRC, and not with the author of this report.

COMPUTING OPTIMAL LOCALLY CONSTRAINED STEPS[1]

David M. Gay

1. Introduction

Many unconstrained minimization algorithms employ a sequence of quadratic approximations $\varphi_i : \mathbb{R}^n \to \mathbb{R}$, $i = 0,1,2,\ldots$ to the objective function $\varphi^* : \mathbb{R}^n \to \mathbb{R}$. Using them, they determine a sequence $x_0, x_1, x_2, \ldots$ of points which (usually) are ever better approximations to a local minimizer x^* of φ^*. Given the current iterate $x = x_i$ and quadratic model $\varphi = \varphi_i$, these algorithms usually compute the next iterate $x_+ = x_{i+1}$ in one of two ways. Either they determine a Newton step s such that $\varphi(x + x)$ is minimized, then set $x_+ = x + \lambda s$, where $\lambda > 0$ is chosen so that $\varphi^*(x + \lambda s) < \varphi^*(x)$, or they choose a neighborhood $N = N_i$ of 0 and a point $s^* \varepsilon N$ which minimizes $\varphi(x + s)$ over $s \varepsilon N$, and they set $x_+ = x + s^*$, having taken care in choosing N that $\varphi^*(x + s^*) < \varphi^*(x)$. Algorithms of the latter sort have an intuitive appeal: φ often approximates φ^* well only in a neighborhood of x, and these algorithms attempt to achieve the maximum function reduction possible on an educated guess at such a neighborhood. This paper concerns itself with choosing s^* in this sort of algorithm, given x and an N of the reasonable form described below. Since we can expect the quadratic approximation φ to φ^* only to be accurate in a neighborhood $x + N$ of the current iterate x, we shall refer to a point s^* that minimizes $\varphi(x + s)$ subject to $s \varepsilon N$ as an otpimal locally constrained (OLC) step.

Suppose now that we have $g \varepsilon \mathbb{R}^n$ and a symmetric nxn matrix $H \varepsilon \mathbb{R}^{nxn}$ such that $\varphi(x + s) = \varphi^*(x) + g^T s + \frac{1}{2} s^T H s$. (We regard vectors as column vectors and use superscript T for "transpose". Superscript $-T$ means "inverse transpose". Of course, we have in mind that $g \doteq \nabla\varphi^*(x)$ and

---

1 Presented at the Tenth International Symposium on Mathematical Programming, Montreal, Canada, 27-31 August 1979.

Sponsored by the United States Army under Contract No. DAAG29-75-C-0024. This material is based upon work supported by the National Science Fouancation under Grant Nos. MCS78-09525 and MCS76-00324.

$H \doteq \nabla^2\varphi^*(x)$ in a suitable sense. We lose no generality in assuming $x = 0$ and $\varphi^*(x) = 0$, so that

$$\varphi(s) = g^T s + \frac{1}{2} s^T H s. \tag{1.1}$$

For the rest of this paper we assume that N has the form

$$N = \{y \in \mathbb{R}^n : \|Dy\| \leq \delta\}, \tag{1.2}$$

where $D \in \mathbb{R}^{nxn}$ is nonsingular, $\|\cdot\|$ denotes the Euclidean norm $\|\cdot\|_2$ (i.e., $\|y\| = (y^T y)^{1/2}$), and $\delta > 0$. Some of the algebra below is simpler if D is the identity matrix I. It is possible, in effect, to arrange this by a change of variables. Let

$$\tilde{g} = D^{-T} g \tag{1.3a}$$

$$\tilde{H} = D^{-T} H D^{-1}, \tag{1.3b}$$

$$\tilde{N} = \{y \in R^n : \|y\| \leq \delta\}, \tag{1.3c}$$

and

$$\tilde{\varphi}(\tilde{s}) = \tilde{g}^T \tilde{s} + \frac{1}{2} \tilde{s}^T \tilde{H} \tilde{s}. \tag{1.3d}$$

If

$$\tilde{s} = Ds, \tag{1.3e}$$

then $\varphi(s) = \tilde{\varphi}(\tilde{s})$ and $s \in N$ if and only if $\tilde{s} \in \tilde{N}$, so in proofs we may just as well deal with $\tilde{\varphi}$ and $\tilde{N}$, for which $\tilde{D} = I$, as with φ and N. Therefore we assume without loss of generality in the proofs below that $D = I$. It is often useful in practice to choose $D \neq I$, e.g. to reflect scale in the

components of s, so we leave D in the statements of some results.

Some of the proofs below are simpler if H is a diagonal matrix. We may assume this without losing generality, because we can also arrange for it, in effect, by changing variables. Specifically, since $\tilde{H}$ is symmetric, there exists an orthogonal matrix $V \in \mathbb{R}^{n \times n}$ such that $\tilde{\tilde{H}} = V\tilde{H}V^T$ is diagonal with nonincreasing diagonal elements. Thus if $\tilde{\tilde{g}} = V\tilde{g}$, $\tilde{\tilde{\varphi}}(\tilde{\tilde{s}}) = \tilde{\tilde{g}}^T\tilde{\tilde{s}} + \frac{1}{2}\tilde{\tilde{s}}^T\tilde{\tilde{H}}\tilde{\tilde{s}}$, and $\tilde{\tilde{s}} = V\tilde{s}$, then $\tilde{\tilde{\varphi}}(\tilde{\tilde{s}}) = \tilde{\varphi}(\tilde{s})$ and $\tilde{\tilde{s}} \in \tilde{N}$ if and only if $\tilde{s} \in \tilde{N}$.

In many applications H is positive definite. In others, however, H may have one or more nonpositive eigenvalues. This may happen, for instance, when H comes from the "augmented model" in the NL2SOL algorithm [DenGW79]. We therefore consider the general case where H may be indefinite. The scheme we are discussing thus provides an appealing way to deal with negative curvature.

In the next section we show that an OLC step s^* satisfies $(H + \alpha D^T D)s = -g$ for some $\alpha \geq 0$ such that $H + \alpha D^T D$ is positive semidefinite. Our treatment differs from that of Goldfeld, Quandt, and Trotter [GolQT66] in that we consider the special case where $H + \alpha D^T D$ is (nearly) singular. This case requires special handling, which we discuss in §3. In §4 we give a complete algorithm for computing s^*, and we discuss numerical experience with it in the context of NL2SOL in §5.

The present work builds on that of many others. Among the first papers to consider computing an OLC step was that of Marquardt [Mar63], in which φ^* was a nonlinear sum of squares. (The paper of Levenberg [Lev44] that is often mentioned in the same breath actually considers a somewhat different step, one that minimizes $\varphi(x + s) + \|Ds\|^2$.) Goldfeld, Quandt, and Trotter considered a general φ^* but restricted themselves to the case where $H + \alpha D^T D$ is positive definite. Hebden [Heb73] gave an interesting algorithm that computes a

good approximation to s^* when $H + \alpha D^T D$ is sufficiently positive definite and that otherwise computes what often would be a reasonable step. Moré [Mor78] refined Hebden's scheme and specialized it to a good algorithm for the case where φ^* is a sum of squares. The algorithm we give in §4 incorporate Moré's refinements of Hebden's scheme along with a new, often more reasonable way to handle the case of a (nearly) singular $H + \alpha D^T D$. When specialized to least-squares problems, the new algorithm computes the same step as does Moré's, but may expend less work in the special case.

2. Characterizing s^*

The following characterization of an OLC step s^* lies at the heart of the algorithm in §4 for approximating s^*.

Theorem 2.1 If φ and N are given by (1.1) and (1.2), then $s^* \in N$ minimizes $\varphi(s)$ over $s \in N$ if and only if there exists $\alpha^* \geq 0$ such that $H + \alpha^* D^T D$ is positive semidefinite and

$$(H + \alpha^* D^T D)s^* = -g, \tag{2.1}$$

with $\|Ds^*\| = \delta$ if $\alpha^* > 0$. If $g \neq 0$ or H has a negative eigenvalue, then α^* is unique.

Proof: Without loss of generality, $D = I$, (2.1) has the form

$$(H + \alpha^* I)s^* = -g, \tag{2.2}$$

and $H = \text{diag}(h_1, \ldots, h_n)$, with $h_1 \geq h_2 \geq \ldots \geq h_n$.

If $g = (g_1, \ldots, g_n)^T$ has $g_i = 0$ for $i < n$, then the theorem is easily seen to hold, so assume $g_i \neq 0$ for at least one $i < n$.

(Only if): Suppose s^* minimizes φ over N. By the second-order necessary conditions in Theorem 4 of [McC76] there exists $\alpha^* \geq 0$ such that (2.2) holds and either $\alpha^* = 0, \|s^*\| \leq \delta$, and H is positive semidefinite, or else $\alpha^* > 0$, $\|s^*\| = \delta$, and

$$y^T(H + \alpha^* I)y \geq 0 \quad \text{for all } y \in \mathbb{R}^n \text{ with } y^T s^* = 0. \tag{2.3}$$

The "only if" assertion thus holds if $\alpha^* = 0$, so assume $\alpha^* > 0$. In this case, (2.3) implies that $H + \alpha^* I$ has at most one negative eigenvalue, so $h_i \geq -\alpha^*$ for $i < n$ and we must show that $h_n \geq -\alpha^*$. This is clearly so if $h_n = h_{n-1}$,

so assume $h_n < h_{n-1}$. For $\alpha \geq \alpha^*$, define $s(\alpha) \in \mathbb{R}^n$ by

$$s_i(\alpha) = \begin{cases} 0 \text{ if } h_i + \alpha = h_i + \alpha^* = 0 \text{ with } i < n; \\ -g_i/(h_i + \alpha) \text{ if } i < n \text{ and } h_i + \alpha > 0; \\ -\text{sign}(g_n)\left[\delta^2 - \sum_{j<n} (s_j(\alpha))^2\right]^{1/2} \text{ if } i = n. \end{cases}$$

If $h_i + \alpha^* = 0$ with $i < n$, then (2.2) implies $g_i = 0$, so it is reasonable to set $s_i(\alpha^*) = 0$ in this case. Indeed, since s^* minimizes φ on N, it is readily verified that $s_i^* = 0$ if $h_i + \alpha^* = 0$ and then that $s^* = s(\alpha^*)$, provided that $\text{sign}(g_n)$ is properly chosen when $g_n = 0$. Let $\psi(\alpha) = \varphi(s(\alpha))$. Then $\psi(\alpha^*) = \varphi(s^*)$. Now if $h_n + \alpha^*$ were negative, then there would clearly exist $\alpha^\# \geq -h_n > \alpha^*$ such that (2.2) held with s^* replaced by $s^\# = s(\alpha^\#)$. But $\|s^\#\| = \delta$ and it is easily verified that $\psi'(\alpha) < 0$ for $\alpha^* \leq \alpha < \alpha^\#$, so $s^\#$ would be a point in N with $\varphi(s^\#) < \varphi(s^*)$. This inequality (which also follows from Theorem 7.1 of [Gan78]) would contradict the choice of s^*. Thus $h_n + \alpha^*$ must be nonnegative and $H + \alpha^* I$ is positive semidefinite.

(If): Since N is compact, there exist an s^* and α^* of the sort just considered. To complete the proof, we study the extent to which they are determined by (2.2). Therefore we now redefine $s:[0,\infty) \to \mathbb{R}^n$ to be an arbitrary function such that

$$(H + \alpha I)s(\alpha) = -g. \tag{2.4}$$

Let k be such that $h_i > h_n$ for $i < k$ and $h_i = h_n$ for $i \geq k$, and for $y = (y_1,\ldots,y_n)^T$, let $Py = (0,\ldots,0,y_k,\ldots,y_n)^T$. For $\alpha > -h_n$, $s(\alpha) = -(H + \alpha I)^{-1}g$ is uniquely determined by (2.4) and $\|s(\alpha)\|$ decreases strictly monotonically. On the other hand, if $\alpha = -h_n$, then (2.4) can hold

only if $Pg = 0$, in which case there are many choices for $s(-h_n)$. No matter which choice is made, $\|s(-h_n)\| \geq \lim_{\alpha \to (-h_n)^+} \|s(\alpha)\|$, with equality only if $Ps(-h_n) = 0$. Thus for $g \neq 0$ there can be at most one choice of $\alpha \geq -h_n$ such that (2.4) holds with $\|s(\alpha)\| = \delta$. Although α can have any nonnegative value if $g = 0$ and $h_n \geq 0$, α is uniquely determined if $g = 0$ with $h_n < 0$, in which case (2.4) can only hold with $\|s(\alpha)\| = \delta$ if $\alpha = -h_n$. This establishes the claims about the uniqueness of α^*.

Suppose now that (2.2) holds with $\alpha^* \geq -h_n$ and that either $\alpha^* = 0$ and $\|s^*\| \leq \delta$ or else $\alpha^* > 0$ and $\|s^*\| = \delta$. If $\alpha^* > -h_n$, i.e. if $H + \alpha I$ is positive definite, then s^* is uniquely determined by (2.2) and is hence the minimizer of φ on N. Assume therefore that $\alpha^* = -h_n$. Then (2.2) uniquely determines s_i^* for $i < k$ and implies $Pg = 0$. Clearly $\varphi(s) = \varphi(s^*)$ for any choice of s having $s_i = s_i^*$ for $i < k$ and $(\|s\| - \delta)\alpha^* = 0$. Since we have seen from (2.2) and the uniqueness of α^* (for $g \neq 0$) that any minimizer of φ on N is such an s, s^* minimizes φ on N. ■

3. (Near) Singularity in $H + \alpha^* D^T D$

When $g \neq 0$ (as we henceforth assume), it is usually possible to compute an approximate OLC step s as

$$s = s(\alpha) = -(H + \alpha D^T D)^{-1} g, \tag{3.1}$$

where $\alpha \geq 0$ is chosen so that $H + \alpha D^T D$ is positive definite and $\| Ds(\alpha) \|$ is near δ. Such a step s exactly minimizes φ (given by (1.1)) on an approximation to the "trust region" N given by (1.2) If $H + \alpha^* D^T D$ is singular in (2.1), however, then it may be impossible to compute a suitable s in this way, because it can happen that $\| Ds(\alpha) \|$ is considerably less than δ for all α that make $H + \alpha D^T D$ positive definite. And if $H + \alpha^* D^T D$ is nearly singular, then computing a sufficiently large $s(\alpha)$ may be impractical or at least unduly costly. Of course, we could simply accept a "short" $s(\alpha)$. But if H has a significantly negative eigenvalue and is a good approximation to the current true Hessian $\nabla^2 \varphi^*(x)$, then it may prove well worthwhile to compute a step $\hat{s}$ having a significant component in the direction of an eigenvector of H corresponding to the smallest eigenvalue. In what follows we describe a simple way to compute such a step $\hat{s}$. This step approximately minimizes φ (to within a prescribed tolerance) on the exact trust region N of (1.2).

To simplify the notation, we assume for the rest of this section that $D = I$ and $H = \mathrm{diag}(h_1, \ldots, h_n)$ with $h_1 \geq h_2 \geq \ldots \geq h_n$, as in the proof of Theorem 2.1. We also assume $h_n \leq 0$.

When $h_n \leq 0$, the algorithm of § 4 maintains a lower bound η on acceptable values of α such that $\eta \leq -h_n$. If $\alpha > -h_n$ yields $\| s(\alpha) \| < \delta$, then

$$\eta \leq -h_n \leq \alpha^* \leq \alpha. \tag{3.2a}$$

It will be convenient to let

(3.2b) $\theta = \alpha - \eta$.

We may regard $H + \alpha^* I$ as nearly singular if we encounter an $\alpha > -h_n$ for which θ is sufficiently small (in the sense made precise in Theorem 3.2 below) and for which $\| s(\alpha) \|$ is unacceptably small, say

(3.3) $\| s(\alpha) \| < \beta\delta$,

for some prescribed $\beta \in (0,1)$, e.g. $\beta = 0.75$ or $\beta = 0.9$.

Suppose now that (3.3) holds, and let $s = s(\alpha)$. For $h_i > h_n$, we may expect that $s_i \doteq s_i^*$, while for $h_i \doteq h_n$, we may expect $g_i \doteq 0$. Thus it seems reasonable to consider computing an approximate OLC step $\hat{s}$ by finding an approximate eigenvector v of H corresponding to h_n and adding a suitable multiple of v to s. Because of (3.2), $H + I$ has an eigenvalue $h_n + \alpha$ with $0 \leq h_n + \alpha \leq \alpha - \eta = \theta$ (and eigenvectors corresponding to this eigenvalue are also eigenvectors of H corresponding to h_n). We probably have a factorization of $H + \alpha I$, so for $\kappa > 1$ (e.g. $\kappa \approx 2$), it should be easy to compute $v \in \mathbb{R}^n$ by the inverse power method such that

(3.4a) $\| v \| = 1$ and

(3.4b) $\| (H + \alpha I)v \| \leq \kappa\theta$.

Given such a v, we may compute

(3.5a) $\hat{s} = s + \sigma v$,

with σ chosen so that $\| \hat{s} \| = \delta$ and $\varphi(\hat{s})$ is minimized. Specifically:

Lemma 3.1: If $\psi(\tau) = \varphi(s + \tau v)$, and if

(3.5b) $\sigma = (\delta^2 - \| s \|^2) / [s^T v + \mathrm{sign}(s^T v)\sqrt{(s^T v)^2 + \delta^2 - \| s \|^2}\ ]$,

(with $\text{sign}(s^T v) = 1$ or -1 if $s^T v = 0$), then $\tau = \sigma$ minimizes $\psi(\tau)$ subject to the constraint $\|s + \tau v\| = \delta$.

Proof: There are two values of τ for which $\|s + \tau v\| = \delta$, namely

$$\tau_+ = -s^T v + \sqrt{(s^T v)^2 + \delta^2 - \|s\|^2} \quad \text{and} \tag{3.6a}$$

$$\tau_- = -s^T v - \sqrt{(s^T v)^2 + \delta^2 - \|s\|^2}\,. \tag{3.6b}$$

Because of (1.1) and (3.1),

$$\begin{aligned}\psi(\tau) &= \varphi(s) + \tau g^T v + \tau s^T H v + \tfrac{1}{2}\tau^2 v^T H v \\ &= \varphi(s) + \tau [g + (H + \alpha I)s]^T v - \tau\alpha s^T v + \tfrac{1}{2}\tau^2 v^T H v \\ &= \varphi(s) - \tau\alpha s^T v + \tfrac{1}{2}\tau^2 v^T H v.\end{aligned}$$

Using (3.6) and (3.4a), we find

$$\begin{aligned}\psi(\tau_+) - \psi(\tau_-) &= -(\tau_+ - \tau_-)\alpha s^T v + \tfrac{1}{2}(\tau_+ - \tau_-)(\tau_+ + \tau_-) v^T H v \\ &= -(\tau_+ - \tau_-)(s^T v)(\alpha + v^T H v) \\ &= -2\sqrt{(s^T v)^2 + \delta^2 - \|s\|^2}\;(s^T v) v^T (H + \alpha I) v.\end{aligned}$$

But $H + \alpha I$ is positive definite, so $v^T(H + \alpha I)v > 0$ and $\psi(\tau_+) > \psi(\tau_-)$ if and only if $s^T v < 0$. Thus the choice $\tau = -s^T v + \text{sign}(s^T v)\sqrt{(s^T v)^2 + \delta^2 - \|s\|^2}$ minimizes ψ subject to $\|s + \tau v\| = \delta$, which is equivalent to $\tau = \sigma$ with σ given by (3.5b). ■

We now consider how to tell whether the θ of (3.2) is small enough that the relative difference between $\varphi(s^*)$ and $\varphi(\hat{s})$ is small, i.e., that

(3.7) $$\epsilon\varphi(\hat{s}) \leq \varphi(s^*) - \varphi(\hat{s}) \leq 0$$

for some prescribed $\epsilon \in (0,1)$, e.g. $\epsilon = 0.1$. To this end, it is convenient to define $\tilde{s} : [\alpha^*, \infty) \to \mathbb{R}^n$ by

(3.8) $$\tilde{s}_i(\tau) = \begin{cases} s_i(\tau) = -g_i/(h_i + \tau) & \text{if } i < n; \\ -\operatorname{sign}(g_n)\sqrt{\delta^2 - \sum_{j<n} s_j(\tau)^2} & \text{if } i = n. \end{cases}$$

In the event that $h_i = h_n$ for some $i < n$, we assume without losing generality that $g_i = s_i^* = 0$ and interpret (3.8) as specifying $\tilde{s}_i(\alpha^*) = 0$. If $g_n = 0$, then we assume that $\operatorname{sign}(g_n) = -\operatorname{sign}(s_n^*)$ in (3.8). Thus, in all cases $\tilde{s}$ is a smooth function with $\tilde{s}(\alpha^*) = s^*$ and $\|\tilde{s}\| \equiv \delta$.

To bound $\varphi(\hat{s}) - \varphi(s^*)$, we shall first bound $\varphi(\tilde{s}(\alpha)) - \varphi(s^*)$, then bound $\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha))$. To bound $\varphi(\hat{s}) - \varphi(s^*)$, it is convenient to define $\psi : [\alpha^*, \infty) \to \mathbb{R}$ by

(3.9) $$\psi(\tau) := \varphi(\tilde{s}(\tau)) - g_n\tilde{s}_n(\tau)$$

$$= \sum_{i<n}\left(\frac{-g_i^2}{h_i + \tau} + \frac{h_i g_i^2}{2(h_i+\tau)^2}\right) + \frac{1}{2} h_n\left(\delta^2 - \sum_{i<n} \frac{g_i^2}{(h_i+\tau)^2}\right)$$

$$= \frac{1}{2} h_n\delta^2 - \frac{1}{2}\sum_{j<n} g_i^2(2\tau + h_i + h_n)/(h_i+\tau)^2.$$

Note that $\psi'(\tau) = \sum_{i<n} g_i^2 (h_n+\tau)/(h_i+\tau)^3 \leq \sum_{i<n} s_i^{*2} (h_n+\tau)/(h_i+\tau) \leq \sum_{i<n} s_i^{*2} \leq \delta^2$.

Since $\alpha - \alpha^* \leq \theta$ by (3.2), we thus find

(3.10) $$\psi(\alpha) - \psi(\alpha^*) \leq \theta\delta^2.$$

Now (3.2) and (3.3) imply

(3.11) $$|g_n| \leq (h_n + \alpha)\|s\| \leq \theta\beta\delta,$$

while (3.8) implies $|\tilde{s}_n(\alpha^*) - \tilde{s}_n(\alpha)| \leq \delta$. Together with (3.9) and (3.10), these yield

(3.12) $$\begin{aligned}\varphi(\tilde{s}(\alpha)) - \varphi(s^*) &= \psi(\alpha) - \psi(\alpha^*) - g_n[\tilde{s}_n(\alpha^*) - \tilde{s}_n(\alpha)] \\ &\leq (1+\beta)\theta\delta^2.\end{aligned}$$

Now we deduce a bound on $\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha))$. For brevity, denote $\tilde{s}(\alpha)$ by $\tilde{s}$. Then (1.1) gives

(3.13) $$\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha)) = g^T(\hat{s} - \tilde{s}) + \frac{1}{2}\hat{s}^T H\hat{s} - \frac{1}{2}\tilde{s}^T H\tilde{s}.$$

To derive a convenient expression for $\hat{s}^T H\hat{s}$, it is useful to let

(3.14) $$f = (H + \alpha I)v,$$

so that $Hv = f - \alpha v$. Note from (3.4b) that

(3.15) $$\|f\| \leq \kappa\theta.$$

Since $\hat{s} = s + \sigma v$ and $\|\hat{s}\| = \delta$, we have $\sigma^2 + 2\sigma s^T v = \delta^2 - \|s\|^2$ and

$$\begin{aligned}\hat{s}H\hat{s} &= s^T Hs + \sigma^2 vHv + 2\sigma s^T Hv = s^T Hs + \sigma^2(v^T f - \alpha) + 2\sigma(s^T f - \alpha s^T v) \\ &= s^T Hs - \alpha[\delta^2 - \|s\|^2] + \sigma[\sigma v + 2s]^T f.\end{aligned}$$

Similarly, we have $\tilde{s} = s + \tilde{\sigma}e_n$, where $e_n = (0,\ldots,0,1)^T$ is the n-th standard unit vector of $\mathbb{R}^n$ and $\tilde{\sigma}$ is chosen so that $\|\tilde{s}\| = \delta$ (with $\text{sign}(\tilde{s}_n) = \text{sign}(s_n)$), and we find

(3.10) $$\psi(\alpha) - \psi(\alpha^*) \leq \theta\delta^2.$$

Now (3.2) and (3.3) imply

(3.11) $$|g_n| \leq (h_n + \alpha)\|s\| \leq \theta\beta\delta,$$

while (3.8) implies $|\tilde{s}_n(\alpha^*) - \tilde{s}_n(\alpha)| \leq \delta$. Together with (3.9) and (3.10), these yield

(3.12) $$\begin{aligned}\varphi(\tilde{s}(\alpha)) - \varphi(s^*) &= \psi(\alpha) - \psi(\alpha^*) - g_n[\tilde{s}_n(\alpha^*) - \tilde{s}_n(\alpha)] \\ &\leq (1+\beta)\theta\delta^2.\end{aligned}$$

Now we deduce a bound on $\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha))$. For brevity, denote $\tilde{s}(\alpha)$ by $\tilde{s}$. Then (1.1) gives

(3.13) $$\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha)) = g^T(\hat{s} - \tilde{s}) + \frac{1}{2}\hat{s}^T H\hat{s} - \frac{1}{2}\tilde{s}^T H\tilde{s}.$$

To derive a convenient expression for $\hat{s}^T H\hat{s}$, it is useful to let

(3.14) $$f = (H + \alpha I)v,$$

so that $Hv = f - \alpha v$. Note from (3.4b) that

(3.15) $$\|f\| \leq \kappa\theta.$$

Since $\hat{s} = s + \sigma v$ and $\|\hat{s}\| = \delta$, we have $\sigma^2 + 2\sigma s^T v = \delta^2 - \|s\|^2$ and

$$\begin{aligned}\hat{s}H\hat{s} = s^T Hs + \sigma^2 vHv + 2\sigma s^T Hv &= s^T Hs + \sigma^2(v^T f - \alpha) + 2\sigma(s^T f - \alpha s^T v) \\ &= s^T Hs - \alpha[\delta^2 - \|s\|^2] + \sigma[\sigma v + 2s]^T f.\end{aligned}$$

Similarly, we have $\tilde{s} = s + \tilde{\sigma}e_n$, where $e_n = (0,\ldots,0,1)^T$ is the n-th standard unit vector of $\mathbb{R}^n$ and $\tilde{\sigma}$ is chosen so that $\|\tilde{s}\| = \delta$ (with $\text{sign}(\tilde{s}_n) = \text{sign}(s_n)$), and we find

$$\tilde{s}^T H \tilde{s} = s^T H s + h_n[\delta^2 - \|s\|^2] .$$

Together with (3.13), these equations imply

$$\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha)) = g^T(\hat{s} - \tilde{s}) - \frac{1}{2}(\alpha + h_n)(\delta^2 - \|s\|^2) + \frac{1}{2}\sigma(\sigma v + 2s)^T f$$

$$\leq g^T(\sigma v - \tilde{\sigma} e_n) + \frac{1}{2}\sigma(\sigma v + 2s)^T f .$$

Now (3.1) and (3.14) give $g^T v = -s^T(H + \alpha I)v = -s^T f$, and the definitions of σ and $\tilde{\sigma}$ imply $|\sigma| \leq \delta$ and $|\tilde{\sigma}| \leq \delta$, so (3.3), (3.4a), (3.11), and (3.15) combine with the above inequality to give $\varphi(\hat{s}) - \varphi(\tilde{s}(\alpha)) \leq \beta\delta^2\kappa\theta + \theta\beta\delta^2 + \frac{1}{2}\delta(\delta+2\beta\delta)\kappa\theta$

$$\leq [\beta(2\kappa + 1) + \frac{1}{2}\kappa]\theta\delta^2 .$$

Combining this with (3.12) and Theorem 2.1, we finally obtain

$$0 \leq \varphi(\hat{s}) - \varphi(s^*) \leq [2\beta(\kappa + 1) + \frac{1}{2}\kappa + 1]\theta\delta^2 . \tag{3.16}$$

This leads to

Theorem 3.2: Let $\beta \in (0,1)$, $\epsilon \in (0,1)$, and $\kappa \in (1,\infty)$ be given and suppose $g \neq 0$ and that $\alpha \neq 0$ renders $H + \alpha I$ positive definite. If the s of (3.1) satisfies (3.3) and the θ of (3.2) satisfies

$$\theta \leq \epsilon\, \delta^{-2}(\alpha\|s\|^2 - g^T s)/[4\beta(\kappa + 1) + \kappa + 2] , \tag{3.17}$$

and if $\hat{s}$ is determined by (3.5), where v satisfies (3.4), then (3.7) holds.

Proof: From (1.1) and (3.1) we have

$$\varphi(s) = g^T s + \frac{1}{2} s^T(H + \alpha I)s - \frac{1}{2}\alpha\|s\|^2$$

$$= g^T s - \frac{1}{2} g^T s - \frac{1}{2}\alpha\|s\|^2$$

$$= \frac{1}{2}(g^T s - \alpha\|s\|^2) .$$

Together with (3.16), (3.17), and the fact that $\varphi(s) < 0$, this implies $\epsilon\varphi(s) \leq \varphi(s^*) - \varphi(\hat{s})$. But Lemma 3.1 implies $\varphi(\hat{s}) \leq \varphi(s)$, so (3.7) follows. ■

4. Choosing α

If either H is indefinite or the Newton step $s^{(N)} = -H^{-1}g$ is too large, then computing an OLC step s^* requires finding a solution α^* to the scalar nonlinear equation $\| D(H + \alpha D^T D)^{-1} g \| = \delta$. There are many iterations for approximating such an α^* --see Gander's excellent discussion in §6 of [Gan78]. In view of the fast convergence reported by Moré [Mor78], we prefer to use an iteration proposed by Reinsch [Rei71] and independently by Hebden [Heb73], together with Moré's (modification of Hebden's) safeguarding scheme. Let

$$\psi(\alpha) := \| D(H + \alpha D^T D)^{-1} g \|^{-1} - \delta^{-1} . \tag{4.1}$$

The basic iteration is Newton's method applied to ψ. Thus if iterate α_k renders $H + \alpha_k D^T D$ positive definite but yields an unacceptable step, then we compute a tentative value $\bar{\alpha}_{k+1} = \alpha_k - \psi(\alpha_k)/\psi'(\alpha_k)$ for α_{k+1}, i.e.

$$\bar{\alpha}_{k+1} = \alpha_k + \| Ds \|^2 (\| Ds \| - \delta) / [\delta s^T D^T D (H + \alpha D^T D)^{-1} D^T D s] , \tag{4.2a}$$

where

$$s = s(\alpha_k) = -(H + \alpha_k D^T D)^{-1} g . \tag{4.2b}$$

We also maintain lower and upper bounds ℓ_k and u_k on α^* and, if H is not positive definite, a value $\eta_k \geq 0$ such that $-\eta_k$ is an upper bound on the smallest eigenvalue of $D^{-T} H D^{-1}$. (For convenience, we set $\eta_k = -1$ if H is positive definite.) We discuss below how these quantities are updated. Once ℓ_{k+1} and u_{k+1} have been determined, we obtain a safeguarded α_{k+1} from the rule

$$\alpha_{k+1} = \begin{cases} \bar{\alpha}_{k+1} & \text{if } \ell_{k+1} \leq \bar{\alpha}_{k+1} < u_{k+1} \text{ and } \bar{\alpha}_{k+1} > \eta_{k+1} ; \\ \max \{ 10^{-3} \cdot u_{k+1}, (\ell_{k+1} u_{k+1})^{1/2} \} & \text{otherwise.} \end{cases} \tag{4.3}$$

Again assume variables have been changed so that $D = I$. To solve linear systems involving $H + \alpha_k I$, e.g. to compute $s(\alpha_k)$ in (3.1), we recommend attempting to compute the Cholesky (or LDL^T) decomposition of $H + \alpha_k I$ (see e.g. §3.3 of [Ste73]). If this works, then we may regard $H + \alpha_k I$ as numerically positive definite (provided the Cholesky factor has no zeros on the diagonal). Otherwise for some ℓ between 1 and n we may express the leading principal $\ell \times \ell$ submatrix of $H + \alpha_k I$ as LML^T, where L is a lower triangular matrix with nonzero diagonal and M is the diagonal matrix $\mathrm{diag}(1, 1, \ldots, 1, \mu)$. Both L and μ are readily available as a byproduct of the attempted factorization, and $\mu \leq 0$. We compute $z = L^{-T} e_\ell$ (i.e., solve $L^T z = e_\ell$), where $e_\ell = (0, 0, \ldots, 0, 1,)^T \in \mathbb{R}^\ell$. Then $\begin{pmatrix} z \\ 0 \end{pmatrix}^T (H + \alpha_k I) \begin{pmatrix} z \\ 0 \end{pmatrix}$ $= z^T L M L^T z = e_\ell^T M e_\ell = \mu$, so $\mu / \| z \|^2$ is a Raleigh quotient for $H + \alpha_k I$, and $(\mu / \| z \|^2) - \alpha_k \leq 0$ is an upper bound on the smallest eigenvalue of H. Hence we set

$$\eta_{k+1} = \ell_{k+1} = \alpha_k - \mu / \| z \|^2 \quad \text{and} \tag{4.4a}$$

$$u_{k+1} = u_k \tag{4.4b}$$

and choose $\bar{\alpha}_{k+1} = \alpha_k$ (which will force a safeguarded choice of α_{k+1} in (4.3)).

If $H + \alpha_k I$ is positive definite then the concavity of ψ [Rei71] implies that the Newton iterate $\bar{\alpha}_{k+1}$ given by (4.2) satisfies $\bar{\alpha}_{k+1} \leq \alpha^*$. In this case we recommend using the following variation on Moré's update prescription for ℓ_k and u_k: if $\psi(\alpha_k) < 0$, then choose

$$\ell_{k+1} = \bar{\alpha}_{k+1} \quad \text{and} \tag{4.5a}$$

$$u_{k+1} = u_k \,. \tag{4.5b}$$

Otherwise choose

(4.6a) $$\ell_{k+1} = \max\{\ell_k, \bar{\alpha}_{k+1}\} \qquad \text{and}$$

(4.6b) $$u_{k+1} = \alpha_k .$$

In both cases, let $\eta_{k+1} = \eta_k$.

To obtain the initial bounds ℓ_1 and u_1, we obtain lower and upper bounds EMIN and EMAX on the eigenvalues of H from the Gerschgorin circle theorem (optimized by the diagonal scaling technique described below), and we exploit the following observation: the α^* of Theorem 2.1 is such that $\delta = \|g\|/(\lambda + \alpha^*)$ for some λ between the smallest and largest eigenvalues of H. Thus $(\|g\|/\delta) - \text{EMAX} \leq \alpha^* \leq (\|g\|/\delta) - \text{EMIN}$, and we let

(4.7a) $$\ell_1 = \max\{\ell_o, (\|g\|/\delta) - \text{EMAX}\} ,$$

(4.7b) $$u_1 = (\|g\|/\delta) - \text{EMIN} ,$$

where ℓ_o is given by whichever of (4.4a) or (4.5a) applies, with $\alpha_{-1} = \ell_{-1} = 0$.

To compute EMIN and EMAX, we use a special case of the scaling by diagonal matrices considered in [Var65]. Specifically, we find a diagonal matrix $\mathcal{D}$ having n-1 diagonal entries of unity and one other positive diagonal entry such that the Gerschgorin lower bound on the spectrum of $\mathcal{D}H\mathcal{D}^{-1}$ is as large as possible, and we use this bound as EMIN. This is quickly done (in $O(n^2)$ operations) as follows: Compute the off-diagonal row sums

(4.8a) $$\sigma_j = \sum_{i \neq j} |H_{ij}|$$

and find k such that row k gives the minimum Gerschgorin lower bound:

(4.8b) $$H_{kk} - \sigma_k = \min\{ H_{jj} - \sigma_j \mid 1 \leq j \leq n\} .$$

For $j \neq k$, let θ_j denote the auxilliary quantity

(4.8c) $$\theta_j = (H_{kk} - H_{jj} + \sigma_j - |H_{jk}|)/2$$

and compute

(4.8d) $$\text{EMIN} = H_{kk} - \max \{\theta_j + (\theta_j^2 + \sigma_k |H_{jk}|)^{1/2} \mid j \neq k\} .$$

EMAX is computed similarly: with σ_j as in (4.8a), find k such that

(4.9b) $$H_{kk} + \sigma_k = \max\{ H_{jj} + \sigma_j \mid 1 \leq j \leq n\} .$$

For $j \neq k$, let

(4.9c) $$\theta_j = (H_{jj} - H_{kk} + \sigma_j - | H_{jk}|)/2$$

and compute

(4.9d) $$\text{EMAX} = H_{kk} + \max\{ \theta_j + (\theta_j^2 + \sigma_k | H_{jk}|)^{1/2} \mid j \neq k\} .$$

We begin the quest for α^* by trying $\alpha = 0$. If this proves unsatisfactory, then we compute ℓ_1 and u_1 by (4.7-9). If acceptable values $\alpha^{(\text{prev})}$ and $\delta^{(\text{prev})}$ of α and δ from a previously computed OLC step are available, then we obtain $\bar{\alpha}_1$ from a rule which, according to J. E. Dennis [private communication], J.J. Moré has found helpful:

(4.10) $$\bar{\alpha}_1 = \delta^{(\text{prev})} \alpha^{(\text{prev})} / \delta .$$

If $\alpha^{(\text{prev})}$ and $\delta^{(\text{prev})}$ are unavailable, then we simply set $\bar{\alpha}_1 = 0$.

To prevent excessive iterations, we deem the step $s = s^k$ computed from α_k acceptable if $\beta\delta \leq \|s^k\| \leq \gamma\delta$ for some specified $\beta \in (0,1)$ and $\gamma \in (1,\infty)$. (Hebden [Heb73] and Moré [Mor78] choose $\beta = 0.9$ and $\gamma = 1.1$. In connection with an algorithm like that of NL2SOL [DenGW79], where $\delta/\| s^{(\text{prev})} \|$ or $\delta/\delta^{(\text{prev})}$

either equals unity or two or lies in [0.1, 0.5], Dennis and Schnabel suggest $\beta = 0.75$ and $\gamma = 1.5$ [DenS79]. Our computational experience with NL2SOL slightly favors the former choice.)

In practice, D is usually a diagonal matrix, so the explicit change of variables (1.3) is easily performed, and we recommend actually performing it when H is given explicitly. When H has the form J^TJ, g has the form J^Tr, and J and r are given explicitly, on the other hand, we prefer the technique advocated by Moré [Mor78], i.e., using a QR factorization of $\begin{bmatrix} J \\ \alpha D^{1/2} \end{bmatrix}$ to compute s.

The statement of Algorithm 4.1 below involves fewer tildas if the notation of (1.3) is reversed, so that the given assignment is finding $\tilde{s}^*$ to minimize $\tilde{\varphi}(\tilde{s}) := \tilde{g}^T\tilde{s} + \frac{1}{2}\tilde{s}^T\tilde{H}\tilde{s}$. When $\tilde{H}$ is given explicitly and $\tilde{g} \neq 0$, the method described above for computing a reasonable approximation $\tilde{s}$ to $\tilde{s}^*$ may be summarized as follows:

Algorithm 4.1:

Compute $H = D^{-T}\tilde{H}D^{-1}$ and $g = D^{-T}\tilde{g}$.

If H is positive definite, then:

Compute the Newton step $s^{(N)} = -H^{-1}g$.

If $\|s^{(N)}\| \leq \gamma\delta$, then halt and return $\tilde{s} = D^{-1}s^{(N)}$.

Set $\eta_1 = -1$ and determine ℓ_o from (4.5) with $\alpha_{-1} = \ell_{-1} = 0$.

Else [H not positive definite] Compute η_o and ℓ_o from (4.4) with $\alpha_{-1} = \ell_{-1} = 0$ and set $\eta_1 = \eta_o$.

Compute EMIN and EMAX by (4.8) and (4.9), and compute ℓ_1 and u_1 from (4.7).

If $\delta^{(prev)}$ and $\delta^{(prev)}$ are available, compute $\bar{\alpha}_1$ from (4.10); otherwise let $\bar{\alpha}_1 = 0$.

Compute α_1 from (4.3).

For $k = 1, 2, \ldots$

If $H + \alpha_k I$ is positive definite, then

Set $\eta_{k+1} = \eta_k$ and compute $s^{(k)} = -(H + \alpha_k I)^{-1} g$.

If $\beta\delta \leq \|s^{(k)}\| \leq \gamma\delta$, then halt and return $\tilde{s} = D^{-1} s^{(k)}$.

Compute $\bar{\alpha}_{k+1}$ from (4.2) with $D := I$.

If $\|s^{(k)}\| < \beta\delta$, then

If $\eta_k \geq 0$ and $\theta = \alpha_k - \eta_k$ satisfies

(3.17) with $s = s^{(k)}$, then

Compute v satisfying (3.4).

Compute $\hat{s}$ from (3.5).

Halt and return $\tilde{s} = D^{-1}\hat{s}$.

Compute ℓ_{k+1} and u_{k+1} from (4.6).

Else [$\|s^{(k)}\| > \gamma\delta$] compute ℓ_{k+1} and u_{k+1} from (4.5).

Else [$H + \alpha_k I$ not positive definite] set $\bar{\alpha}_{k+1} = \alpha_k$

and compute η_{k+1}, ℓ_{k+1}, u_{k+1} from (4.4).

Determine α_{k+1} from (4.3).

After an OLC step has been computed, it is often necessary to compute another OLC step from the same g and H with a new value of δ. To handle this situation, it is worthwhile to modify the initial part of Algorithm 4.1 to take advantage of the extra information that is available. For example, if δ has been increased, then u_1 can be set to the minimum of the value given by (4.7b) and the last value $u^{(old)}$ of u_k, while if δ has been decreased, ℓ_1 can be set to the maximum of the value given by (4.7a) and the last value $\ell^{(old)}$ of ℓ_k. In either case, η_1 can be set to $\eta^{(old)}$ and $\bar{\alpha}_1$ can be computed from (4.2) with $\alpha_o = \alpha^{(old)}$.

5. Numerical Experience with NL2SOL

For use in NL2SOL [DenGW79], we have implemented two versions of Algorithm 4.1: GQTSTP is designed for use with a general H, while LMSTEP deals explicitly with J (or its QR decomposition) when H has the form J^TJ, g has the form J^Tr, and J is a rectangular matrix with at least as many rows as columns. Both codes make special provision for the case in which an OLC step is to be recomputed with a new δ but the same g and H. To handle D, which both codes assume to be a diagonal matrix, GQTSTP explicitly changes variables, whereas LMSTEP follows the procedure recommended by Moré [Mor78].

The way these codes deal with (near) singularity in $H + \alpha^*D^TD$ deserves some further discussion. Both detect this case by test (3.17). In the case of LMSTEP and true singularity in $H + \alpha^*D^TD$, we would have $\alpha^* = 0$ and H positive semidefinite, and any v in the null space of H would be orthogonal to g, whence $\varphi(s + v) = \varphi(s)$. LMSTEP therefore returns without modification a step s for which (3.3) and (3.17) hold. GQTSTP similarly avoids replacing s by $\hat{s} = s + \sigma v$ in cases where $\varphi(s)$ and $\varphi(\hat{s})$ would not differ significantly. Specifically, if $\varphi(s) - \varphi(\hat{s}) < -(\epsilon/3)\varphi(s)$, then GQTSTP returns s rather than $\hat{s}$. To assure that (3.7) holds in this case, GQTSTP uses (3.17) with ϵ replaced by $2\epsilon/3$.

LMSTEP and GQTSTP depart slightly from Algorithm 4.1 in the calculation of α_{k+1} and ℓ_{k+1}. They use $\alpha_{k+1} = \bar{\alpha}_{k+1}$ if $\ell_{k+1} < \bar{\alpha}_{k+1} < u_{k+1}$ in place of the first part of (4.3), $\ell_{k+1} = \ell_k$ in place of (4.6a), and $\ell_{k+1} = \alpha_k - \bar{\psi}(\alpha_k)/\bar{\psi}'(\alpha_k)$ in place of (4.5a), where $\bar{\psi}(\alpha) = \|D(H + \alpha D^TD)^{-1}g\| - \delta$. This gives a worse bound than the one derived from the ψ of (4.1). J. J.

Moré [private communication] pointed out the superiority of the latter, but budgetary and time constraints kept us from incorporating it into LMSTEP and GQTSTP.

Table I gives some statistics on the performance of LMSTEP and GQTSTP when NL2SOL is run on the problem set described in [DenGW79]. (The statistics were gathered on the Univac 1110 computer at the University of Wisconsin using a preliminary double-precision version of NL2SOL with V(VCONCR) = 0 and other input values at their defaults, with the exceptions described in §7 of [DenGW79].) The column headed "No. of Steps" gives the total number of non-Newton steps computed and the column headed "% of Total" tells what percentage these were of all the steps computed by the module in question. The average value of k when Algorithm 4.1 halted (averaged over the non-Newton steps) appears in the column labelled "k mean," and the maximum such value appears under "k max". In the columns headed "Special Case" are the percentage of non-Newton steps in which (near) singularity in $H + \alpha^* D^T D$ was detected and the mean and maximum final values of k for these steps.

It is fortunate that the mean values of k are so low. Had they been much greater than two in the case of LMSTEP or four in the case of GQTSTP, then it would have been possible to save some time in these modules by preprocessing the input matrices to a sparser form: by reducing J to bidiagonal form in LMSTEP or reducing $D^{-1}HD^{-1}$ to tridiagonal form in GQTSTP (see §§7.5 and 7.1 of [Ste73] and the references cited therein).

After all the effort spent in studying the special case, it is somewhat disappointing to see that GQTSTP never detected this case. It will be interesting to see how often GQTSTP detects it when used in solving other kinds of optimization problems. On the other hand, detecting the special

case appears worthwhile in LMSTEP: the mean k values at which it was detected are much lower than the limit on k that would have to have been imposed and reached in the absence of checking for the special case.

			All (Non-Newton) OLC Steps				Special Case		
Module	β	γ	No. of Steps	% of Total	k mean	k max	% of OLC	k mean	k max
LMSTEP	0.9	1.1	871	74.7	1.54	5	4.1	1.42	3
LMSTEP	0.75	1.5	1032	77.3	1.30	4	2.0	1.67	3
GQTSTP	0.9	1.1	283	61.4	1.95	11	0	--	--
GQTSTP	0.75	1.5	232	54.2	1.91	9	0	--	--

Table I: Statistics from NL2SOL test problems

Acknowledgement. I thank Jorge J. Moré for a helpful discussion on some of the material in §4.

REFERENCES

[DenGW79] Dennis, J.E.; Gay, D.M.; & Welsch, R.E. (1979), "An Adaptive Nonlinear Least-Squares Algorithm", Report TR 77-321 (revised), Computer Science Dept., Cornell Univ., Ithaca, N.Y. 14853.

[DenS79] Dennis, J.E.; & Schnabel, R.B. (1979), "Quasi-Newton Methods for Unconstrained Nonlinear Problems", manuscript.

[Gan78] Gander, W. (1978), "On the Linear Least Squares Problem with a Quadratic Constraint", Technical report, Computer Science Dept., Stanford Univ.

[GolQT66] Goldfeld, S.M.; Quandt, R.E.; & Trotter, H.F. (1966), "Maximization by Quadratic Hill-Climbing", Econometrica 34, pp. 541-551.

[Heb73] Hebden, M.D. (1973), "An Algorithm for Minimization Using Exact Second Derivatives", Report TP515, A.E.R.E. Harwell, Oxfordshire. England.

[Lev44] Levenberg, K. (1944), "A Method for the Solution of Certain Non-linear Problems in Least Squares", Quart. Appl. Math. 2, pp. 164-168.

[Mar63] Marquardt, D.W. (1963), "An Algorithm for Least Squares Estimation of Nonlinear Parameters", SIAM J. Appl. Math. 11, pp. 431-441.

[McC76] McCormick, G.P. (1976), "Optimality Criteria in Nonlinear Programming", pp. 27-38 of Nonlinear Programming, SIAM-AMS Proceedings vol. IX, edited by R.W. Cottle and C.E. Lemke, American Mathematical Society, Providence, R.I.

[Mor78] More, J.J. (1978), "The Levenberg-Marquardt Algorithm: Implementation and Theory", pp. 105-116 of Lecture Notes in Mathematics 630, edited by G.A. Watson, Springer Verlag, Berlin, Heidelberg and New York.

[Rei71] Reinsch, C.H. (1971), "Smoothing by Spline Functions. II", Number. Math. 16, pp. 451-454.

[Ste73] Stewart, G.W. (1973), Introduction to Matrix Computations, Academic Press, New York.

[Var65] Varga, R.S. (1965), "Minimal Gerschgorin Sets", Pacific J. Math. 15, pp. 719-729.

DMG/sr

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 2013	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) COMPUTING OPTIMAL LOCALLY CONSTRAINED STEPS		5. TYPE OF REPORT & PERIOD COVERED Summary Report - no specific reporting period 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) David M. Gay		8. CONTRACT OR GRANT NUMBER(s) MCS78-09525 DAAG29-75-C-0024 MCS76-00324
9. PERFORMING ORGANIZATION NAME AND ADDRESS Mathematics Research Center, University of Wisconsin 610 Walnut Street Madison, Wisconsin 53706		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 5 - Mathematical Programming and Operations Research
11. CONTROLLING OFFICE NAME AND ADDRESS See Item 18 below		12. REPORT DATE October 1979 13. NUMBER OF PAGES 25
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park
North Carolina 27709

National Science Foundation
Washington, D.C. 20550

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Unconstrained optimization, negative curvature

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In seeking to solve an unconstrained minimization problem, one often computes steps based on a quadratic approximation q to the objective function. A reasonable way to choose such steps is by minimizing q constrained to a neighborhood of the current iterate. This paper considers ellipsoidal neighborhoods and presents a new way to handle certain computational details when the Hessian of q is indefinite, paying particular attention to a special case which may then arise. The proposed step computing algorithm provides an

CONTINUED

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ABSTRACT CONTINUED

attractive way to deal with negative curvature. Implementations of this algorithm have proved very satisfactory in the nonlinear least-squares solver NL2SOL.